Peringkasan Multi-dokumen Berita Berbahasa Indonesia menggunakan Conditional Random Fields (CRF)

Angga Auliya Akbar[1], Mira Kania Sabariah, ST., MT.[2], Angelina Prima Kurniati, ST., MT.[3]

Fakultas Informatika, Universitas Telkom, Bandung

[1]anggalie@students.telkomuniversity.ac.id, [2]mirakania@telkomuniversity.ac.id, [3]angelina@telkomuniversity.ac.id

Abstrak

Peringkasan multi-dokumen dibutuhkan agar pencarian informasi dapat berjalan lebih efektif dan efisien. Sistem peringkasan ini menggunakan framework ekstraksi yang telah secara luas digunakan untuk meringkas dokumen, dimana hasil ringkasan merupakan kalimat- kalimat yang telah diekstraksi dari dokumen. Peringkasan dokumen dapat dilakukan dengan melihat nilai informasi suatu kalimat dari dokumen tersebut berdasarkan fitur yang diterapkan pada kalimat yang ada. Fitur tersebut diantaranya adalah fitur linguistik dan statistikal seperti posisi kalimat. Conditional Random Fields (CRF) merupakan salah satu model probabilistik untuk mengatasi segmentasi dan pemberian label pada data sekuens. CRF digunakan untuk mengkombinasikan fitur- fitur untuk mendapatkan model yang akan digunakan untuk menilai tingkat kepentingan kalimat. Fitur yang digunakan adalah basic feature dan complex feature. Sistem akan dinilai akurasinya menggunakan F-Measure, dengan membandingkan ringkasan yang dibuat oleh sistem dengan ringkasan yang dibuat oleh ahli. Hasil pengujian menunjukkan bahwa rata- rata akurasi yang dihasilkan sebesar 62,5% dengan menentukan nilai threshold sebesar 0.45 dimana nilai tersebut digunakan untuk mengklasifikasikan kalimat summary dan non-summary.

Kata kunci : Peringkasan multi-dokumen, Conditional Random Fields, Basic Feature, Complex Feature, F-Measure

Abstract

Multi-document summarization is needed in order information retrieval can be done effectively dan efficiently. This summarization system uses extraction framework that has been widely used to summarize the documents, which is the results of the summarization use sentences that have been extracted from documents. Multi-document summarization can be done by looking at the information value of the sentences from document based on the features that are applied to the existing sentences. The features include the linguistic and statistical feature such as position of the sentence. Conditional Random Fields (CRF) is a probabilistic model to overcome the segmentation and labeling sequence data. CRF is used to combine the features to get a model which is used to predict the value of importance of each sentences. Feature that used are basic feature and complex feature. Accuracy of the system is measured by using F-Measure, by comparing summaries created by the system and summaries created by expert. The test results showed that the average accuracy of the system is around 62,5% with value of threshold set 0.45 where that values used to classificating summary sentences and non-summary sentences.

Keywords : Multi-document summarization, Conditional Random Fields, Basic Feature, Complex Feature, F-Measure

1. Pendahuluan

Melihat era sekarang dimana orang cenderung tidak lagi membaca berita dari majalah atau koran konventional, sebagian besar orang mulai beralih untuk membaca berita dari gadget secara online. Sehingga membuat penyedia jasa berita online semakin banyak dijumpai, ada yang berbasis website atau aplikasi yang semakin mudah untuk diakses dimanapun dan kapanpun. Berita yang disajikan beragam dan tentu saja berita tersebut berasal dari berbagai sumber dan sudut pandang penulis yang berbeda juga. Hal tersebut akan membuat pembaca mencari berita dari berbagai sumber untuk melengkapi informasi yang bisa didapatkan. Terkadang ada berita yang dijelaskan panjang lebar dan terlihat berbeda dari sumber berita sebelumnya namun sebenarnya memiliki informasi yang sama. Sebagai contoh, dengan memanfaatkan mesin pencari atau search engine di Internet hanya dengan mengetikkan suatu topik sebagai query maka puluhan hingga ratusan dokumen yang berhubungan dengan topik tersebut dapat langsung diakses.

Namun, kemudahan pengaksesan dokumen berita ini justru memunculkan kesulitan bagi pengguna untuk mendapatkan informasi yang benar- benar diinginkan dan penting dari setiap dokumen, sebab dengan banyaknya dokumen yang ada maka semakin banyak pula waktu dan usaha yang dibutuhkan untuk membacanya dan menyaring informasi yang diinginkan. Oleh sebab itu, dibutuhkan sebuah solusi berupa peringkasan multi-dokumen sehingga pengguna bisa dengan cepat mendapatkan informasi- informasi penting yang ada dari semua dokumen tersebut tanpa harus membaca satu per satu sehingga pencarian informasi menjadi efektif.

Peringkasan dokumen dapat dilakukan dengan melihat nilai informasi suatu kalimat dari dokumen tersebut berdasarkan fitur yang diterapkan pada kalimat yang ada. Fitur tersebut diantaranya adalah fitur linguistik dan statistikal seperti posisi kalimat. Karena pada saat melakukan peringkasan dokumen sering kali ditentukan dari posisi kalimat itu, yang selanjutnya disebut dengan basic feature. Selain itu ada fitur yang melihat kalimat berdasarkan keterkaitan kalimat itu dengan kalimat lain dan pengaruh kalimat tersebut dalam suatu dokumen, yang selanjutnya disebut complex feature. Dengan mengolah informasi fitur maka akan didapatkan nilai kalimat dari setiap kalimat yang ada dan dapat dibuat model untuk dapat diterapkan di dokumen lain.

Kunci dari sistem peringkasan ini adalah ekstraksi kalimat, yaitu mengambil kalimat yang merepresentasikan isi dokumen dengan mempertahankan bentuk dan tidak mengubah konten dari dokumen. Metode peringkasan ekstraktif bisa diklasifikasikan kedalam dua kelompok, yaitu supervised methods yang mengandalkan pasangan dokumen dan ringkasan yang tersedia, dan untuk peringkasannya adalah dengan mengklasifikasikan kalimat kandidat ke dalam kalimat penting dan tidak penting berdasarkan fitur yang digunakan, sedangkan unsupervised methods bertujuan untuk mengambil kalimat berdasarkan pengelompokan semantik yang diekstrak dari dokumen [3].

Dengan adanya fenomena diatas dan dengan solusi yang ada maka diperlukan adanya penelitian dengan judul "Peringkasan Multi-Dokumen Berita Berbahasa Indonesia Menggunakan Conditional Random Fields".

2. Kajian Pustaka

2.1. Text Mining

Teks mining merupakan salah satu spesifikasi dari ilmu Data Mining yang bertujuan untuk menemukan pola-pola menarik yang belum diketahui sebelumnya dengan menggunakan suatu algoritma atau metode tertentu, pola yang akan ditemukan berasal dari data yang berupa text [8].

Penggunaan teks sebagai data memerlukan beberapa proses tambahan karena karakter teks yang berdimensi besar dan sering mengalami perubahan dari bentuk kata dasarnya, misalnya penambahan imbuhan atau penggunaan kata tidak baku. Untuk itu diperlukan adanya proses textpreprocessing untuk mempersiapkan data teks agar dapat diproses. Umumnya textpreprocessing menggunakan preprocessing berupa stemming, tokenisasi, stopword removal dan case folding[8]. Namun ada juga proses lain yang harus dilakukan, hal itu disesuaikan dengan kasus yang ingin diselesaikan.

2.2. Conditional Random Fields

Conditional Random Fields (CRF) merupakan suatu model probabilistik yang banyak digunakan pada proses segmentasi dan pelabelan suatu sekuen data. Salah satu bentuk dari CRF adalah linear-chain CRF. CRF merupakan metode percampuran antara Hidden Markov Model dan Maximum Entropy Markov Model [5].

CRF mempertahankan kelebihan dari metode supervised dan unsupervised dengan tetap menghindari kekurangan kedua metode tersebut[1]. Dengan mengakuisisi kelebihan dari pemodelan diskriminatif yang tidak dimiliki oleh model generatif dan mengatasi kekurangan pada model generatif seperti permasalahan ketergantungan terhadap asumsi yang tinggi pada Hidden Markov Model (HMM) dan permasalahan biasa label yang terjadi pada Maximum Entropy Markov Model (MEMM) [6].

Metode yang dapat digunakan dalam peringkasan dokumen dibedakan menjadi 2, yaitu

a. Supervised, metode supervised menganggap peringkasan dokumen sebagai 2 kelas masalah yang berbeda dan mengklarifikasikan tiap kalimat secara tersendiri tiap kalimatnya tanpa memperhatikan hubungan antar kalimat yang ada.
b. Unsupervise, metode ini menggunakan heuristic dalam menentukan kalimat yang penting secara langsung.

Dalam CRF, untuk anggota variabel X adalah data yang dapat mempengaruhi hasil, dan anggota variabel Y adalah semua hasil. Dimana terdapat kalimat yang akan dilabeli sebagai $X = (x_1, \dots, x_T)$ dan label yang digunakan untuk melabeli kalimat sebagai $Y = (y_1, \dots, y_T)$. Hasil keluaran dari CRF adalah nilai probabilitas P(X|Y) dapat ditulis

$$- \quad \sum_{,} \quad (\quad , ,) + \sum_{,} \quad (\quad ,) \qquad (2.1)$$

= konstanta normalisasi yang menjumlahkan probabilitas dari semua bagian sekuen.

= bobot dari setiap fungsi transisi (arbitrary feature function) yang mencerminkan ketegasan dari fungsi fitur.

= fungsi fitur acak dari seluruh analisa pada sekuen

= bobot dari setiap fitur node (feature function) yang mencerminkan ketegasan dari fungsi fitur

= fitur fungsi yang berdasarkan nilai dan nilai X

= nilai yang menyatakan kalimat x_i merupakan kalimat ringkasan atau bukan

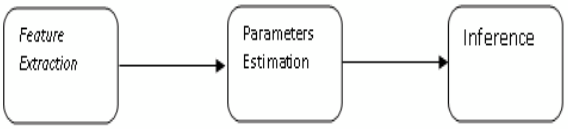
= nilai yang menyatakan kalimat x_{i-1} merupakan kalimat ringkasan atau bukan

x_i = kalimat pada posisi saat itu

x_{i-1} = kalimat sebelumnya

(, ,) = fungsi fitur transisi (arbitrary feature) yang melihat pengaruh label dari kalimat sebelumnya terhadap label kalimat x_i

- State pada posisi dan − 1 ketika (,) merupakan fungsi fitur dari state saat di posisi i dan sekuen pengamatan
- dan adalah bobot optimal yang didapatkan dari proses training dari sekumpulan data latih yang telah memiliki label.
- Secara umum proses CRF ditunjukkan pada Gambar 2.1



Gambar 2.1 Proses didalam CRF

Feature Extraction

Pada proses ini dibuat feature function berdasarkan nilai- nilai fitur yang ada pada setiap kalimat pada data latih. Feature function yang sesuai diekstrak dan menjadi masukan saat perhitungan label.

Contoh feature function atau fitur node yang mungkin didapatkan dari data latih dengan nilai fitur posisinya menurut [1] dinyatakan dalam (,)

$$(,) \begin{cases} 1, & = 1 \quad = 1 \\ 0, & h \end{cases} \quad (2.2)$$

$$(, ,) \begin{cases} 1, & = 1, \ = 0 \quad = 1 \\ 0, & h \end{cases} \quad (2.3)$$

Nilai akan bernilai 1 apabila merupakan kalimat ringkasan (= 1) , kalimat bukan merupakan kalimat ringkasan dan mimiliki nilai fitur posisi = 1 (= 1). Dari contoh pemrosesan fitur diatas dapat dilihat bahwa terdapat 2 jenis fitur yaitu arbitrary feature function atau fitur transisi dan feature function atau fitur node. Perbedaan antara 2 jenis fitur tersebut terletak pada pembuat kondisinya. Nilai fitur transisi akan bernilai 1 tergantung pada label yang ada pada kalimat sebelumnya dan label pada kalimat saat ini. Sedangkan fitur node merupakan fitur yang tidak dipengaruhi oleh nilai dari kalimat lain atau nilai label dari kalimat sebelumnya. Fitur node hanya melihat nilai fitur dan nilai label yang ada pada kalimat tersebut. Pada beberapa studi kasus memungkinkan feature function yang digunakan hanya fitur node saja, namun pada penelitian ini menggunakan kedua jenis fitur, yaitu fitur transisi dan fitur node.

Parameters Estimation

Berdasarkan [12] pada tahap training hal yang dilakukan adalah mendapatkan nilai optimal untuk parameter pendukung feature function yaitu λ. Nilai optimal λ didapatkan dengan menggunakan maximum likelihood yaitu kuantitas yang mencirikan berapa banyak parameter saat ini yang didukung oleh data training. Maximum likelihood ini akan memaksimalkan kemiripan karakteristik antara kumpulan data training yang dimodelkan dengan CRF dengan kumpulan data training itu sendiri. Metode ini umumnya membutuhkan dua perhitungan yakni log-likelihood dan turunan pertamanya pada parameter tertentu. Log-likelihood secara matematis dapat dituliskan dalam persamaan

$$\mathcal{L} = \sum_{[\ ,\ ]} \sum \quad (', ,) - \quad ', ', ,) \log ( ) - \sum \sum_{[\ ,\ ]} \sum \quad ' \quad (2.10)$$

Dimana ( ) adalah kuantitas khusus yang dikenal sebagai fungsi normalisasi berkaitan dengan inputan x. Variabel adalah standar deviasi dari Distribusi Gauss yang nilainya ditetapkan penelitian terkait yang dilakukan sebelumnya yaitu 0.1. Dari persamaan (2.10) maka didapatkan turunan dari log-likelihood feature function ke-k pada data x

Dari gambar tersebut dapat dikatakan bahwa akan memiliki nilai 1, ketika merupakan kalimat ringkasan (memiliki label = 1) (= 1) dan

memiliki nilai fitur posisi 1 (= 1).

Contoh abitrary feature fuction atau fitur

transisi yang mungkin didapatkan dari data latih

dengan melihat label dari kalimat sebelumnya, yang dinyatakan dalam (, ,)

yaitu

$$= \sum_{[,]} ((', ,) -$$

$$\sum (|) (, ,)) - \quad (2.11)$$

$$' = \sum_{[,]} ((,) -$$

$$\sum (, |) (', ' ,)) - \frac{T}{} ' \quad (2.12)$$

dimana (′, ,) adalah feature function dari data training yang telah dilabeli dan (, ,) adalah feature function dari data training yang akan diuji dengan label pada himpunan label yang mungkin.

Akan tetapi, untuk mendapatkan nilai parameter pendukung feature function $_k$ yang optimal, persamaan (2.11) dan (2.12) tidak diselesaikan dengan penyelesaian sistem persamaan linier dikarenakan nilai probabilitas kondisional label untuk data x_t yakni (, |) dan data yang berdampingan (, |) tidak diketahui. Oleh karena itu, untuk menyelesaikan persamaan (2.11) dan (2.12) digunakan metode Stochastic Gradient yang mengimplementasikan pemrograman dinamis dengan memanfaatkan prosedur forward-backward pass untuk menelusuri sekuens data. Penelusuran dengan forward-backward pass bertujuan untuk mendapatkan nilai dari seluruh probabilitas lokal label pada sekuens data. Forward variable pada tiap data x_t secara matematis dapat dituliskan sebagai

$$[\ ] = \sum [\ ]\ (\ ,\ )\ (\ ,\ ,\ ) \quad (2.13)$$

dimana > 0 adalah faktor penskalaan yang besarnya ditentukan sehingga

$$\sum [\ ] = 1.$$

Kemudian, partisi logaritmik dihitung dengan persamaan :

$$\log\ (\ ) = \sum_{[\ ,\ ]} \log\ - \log \sum [\ ]\ (\ ,\ ) \quad (2.14)$$

Adapun backward variable pada tiap data x_t secara matematis dapat dituliskan sebagai

$$[\ ] = \sum [\ ]\ (\ ,\ )\ (\ ,\ ,\ ) \quad (2.15)$$

dimana > 0 adalah faktor penskalaan yang besarnya ditentukan sehingga $\sum [\ ] = 1$.

Setelah semua forward variable dan backward variable untuk tiap data x_t telah memiliki nilai, maka perhitungan probabilitas lokal untuk tiap label yang mungkin pada data x_t dapat dituliskan dalam persamaan matematis sebagai

$$(\ |\ ) = [\ ]\ (\ ,\ )\ (\ ,\ ,\ )\ [\ ] \quad (2.16)$$

$$(\ ,\ |\ ) = \text{Ш}\ [\ ]\ (\ ,\ )\ (\ ,\ ,\ )\ (\ ,\ )\ [\ ] \quad (2.17)$$

dimana adalah faktor normalisasi untuk memastikan bahwa $\sum (\ |\ ) = 1$. Ш adalah faktor normalisasi untuk memastikan bahwa

Setelah nilai probabilitas lokal tiap label yang mungkin untuk tiap data x_t pada sekuens x didapatkan, maka nilai turunan pertama log-likelihood akan didapatkan berdasarkan persamaan (2.17). Kemudian nilai turunan pertama log-likelihood akan digunakan untuk memperbaharui parameter pendukung feature function λ_k pada tiap iterasi berdasarkan persamaan

$$\leftarrow\ +\ \quad (2.18)$$

Inference

Setelah mendapatkan model dari proses training data latih, proses selanjutnya adalah mengimplementasikan model tersebut pada data uji. Data uji merupakan sekumpulan data yang telah memiliki nilai nilai fitur.

Pelabelan secara sekuen yang sesuai dengan kondisional probabilitas (|) dengan parameter ʌ untuk kalimat pada data uji dapat dilakukan menggunakan.

$$^{*} = \ _{\Lambda}(\ |\ ) \quad (2.6)$$

Dimana hal tersebut dapat dikalkulasikan secara efisien oleh Algoritma Viterbi [13]. Algoritma Viterbi digunakan untuk mencari nilai sekuen dari data dengan menelusuri semua sekuens data.

Probabilitas marjinal status pada setiap posisi pada sekuen dapat dihitung menggunakan Viterbi dengan menerapkan prosedur forward-backward, dengan mendefinisikan forward value dengan

$$\sum \sum (\ ,\ |\ ) =$$

1

$$(|) = \sum \quad (|) \quad \wedge (, ,)$$

Dimana $\wedge (, ,)$ didefinisikan oleh

$$\wedge (, ,) = \sum \quad (= \quad , \quad = \quad ,) + \sum \quad (\quad = \quad ,) \qquad (2.8)$$

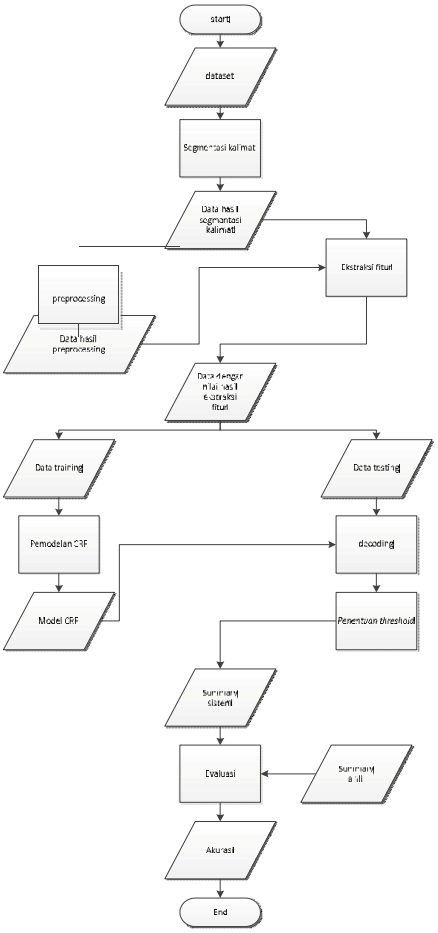
Dengan nilai Backward $(|)$ dapat didefinisikan dengan cara menghitung nilai probabilitas marjinal dari tiap kalimat yang ada menjadi sebuah ringkasan kalimat, dengan menghitung nilai probabilitas semua kalimat sekuen dengan :

$$(\quad = \quad |) = \quad (|) * \quad (|)$$

Lalu dari $(\quad = 1|)$ akan didapatkan nilai probabilitas kalimat– kalimat tersebut merupakan kalimat summary. Nilai probabilitas tersebut di urutkan/diranking lalu ditentukan nilai threshold untuk menentukan kalimat summary dari seluruh dokumen uji.

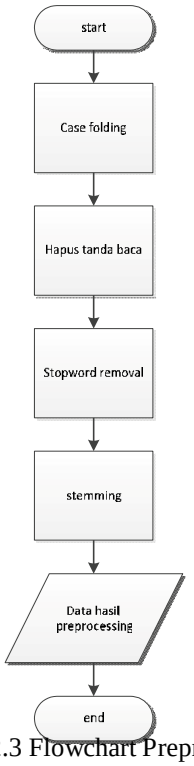
3. Analisis Kebutuhan

3.1 Gambaran Umum



Gambar 2.2 Flowchart Proses Umum Peringkasan Dokumen

3.1. Proprocessing



Gambar 2.3 Flowchart Preprocessing

3.2. Ektraksi Fitur

Fitur merupakan penilaian dari setiap kalimat. Nilai fitur ini diberikan pada kedua jenis data,yaitu data latih dan data uji. Nilai kalimat pada masing-masing featur berbeda, beberapa dari fitur memberikan nilai bolean bernilai 1 atau 0, dan ada pula fitur yang berisi nilai riil. Pada penelitian ini fitur yang diberikan pada setiap kalimat terdiri dari 2 jenis, yaitu Basic Feature dan Complex Feature [1].

Basic Feature merupakan fitur-fitur yang didasarkan pada informasi statistiknya. Beberapa nilai dari fitur ini tidak secara eksplisit merepresentasikan karakteristik dari kalimat. Basic Features yang digunakan dalam penelitian ini adalah :

a. Upper_Case_Words : dalam beberapa kasus, nama orang/tokoh ataupun nama benda akan ditulis dengan huruf besar (Upper Case), dan biasanya nama orang/tokoh atau nama memiliki intisari/makna yang penting.
b. Position : posisi kalimat pada dokumen, bila terletak pada awal kalimat maka "position" bernilai 1, bila pada akhir dokumen bernilai 2, dan pada tengah dokumen bernilai 3.
c. Length : nilai untuk fitur ini berupa integer. Fitur ini menghitung jumlah term yang terkandung dalam x_i setelah menghilangkan stopword.
d. Thematic_Words : kalimat yang mengandung lebih banyak kata thematic (kata yang paling sering muncul) kemungkinan besar adalah summary sentence. Nilai dari setiap kalimat

ditentukan dari jumlah kata thematic pada kalimat tersebut.

e. Log Likelihood : dihitung nilai log probabilitas kemunculan kata

$$\log(\ |\ ) = \sum \quad (\ ,\ ) \qquad \sum \quad , \qquad (2.19)$$

f. Similarity_to_Neighboring_Sentences : untuk menyimpan kemiripan atau similaritas antara kalimat dan kalimat tetangganya, dalam kasus ini menggunakan cosine similarity kalimat x_i dengan 3 kalimat sebelumnya dan 3 kalimat sesudahnya.

$$\cos(\Theta_{ij}) = \frac{\sum_k (d_{ik} d_{jk})}{\sqrt{\sum_k d_{ik}^2} \sqrt{\sum_k d_{jk}^2}}$$

Gambar 2.4 Persamaan Cosine Similarity

Sedangkan untuk Complex Feature, digunakan algoritma yang pada umumnya digunakan untuk metode unsupervise. Fitur tersebut diantaranya,

a. LSA Score
Dengan menguraikan matriks word-sentences melalui Singular Vector Decomposition, nilai ini didapatkan dari matriks A yang berukuran kata x kalimat. Matriks berisi frekuensi kata dalam kalimat. Nilai LSA score didapatkan dari perkalian [11] dimana,

= merupakan nilai vector singular dari sebuah kalimat. Matriks ini merupakan sebuah diagonal matriks, dan nilainya diurutkan dari nilai terbesar.

= disebut right singular vector , merupakan nilai eigen vektor dari . , sehingga matirks berukuran jumlah kalimat x jumlah kalimat.

b. HITS Score
HITS merupakan salah satu algoritma yang biasa digunakan untuk melakukan perankingan terhadap halaman Web berdasarkan pada link yang mengarah ke web tersebut (incoming link) dan link yang mengarah ke web lain (outgoing link). Pada kasus Web, vertex merepresentasikan setiap halaman web. Sehingga graf yang dapat dibentuk oleh vertex tersebut merupakan graf berarah [12].

Nilai HITS dicari dengan membuat matriks keterhubungan kata dengan kalimat, dengan mencari nilai hubs dan authoritynya

$$(\ ) = \sum \in (\ ) \qquad (2.20)$$

Dalam kasus peringkasan dokumen, vertex dapat direpresentasikan oleh kalimat yang ada pada dokumen, dan garis penghubung antar vertex kalimat tersebut merupakan suatu garis yang memiliki bobot yang menyatakan keterhubungan

dari setiap kalimatnya yaitu nilai similaritas antar kalimatnya [13]. Hubungan antar kalimat

tersebut dapat diartikan sebagai suatu kalimat merujuk kepada kalimat lain yang memiliki kesamaan.

3.3. Precision, Recall dan F-Measure

F-Measure digunakan pada tahap evaluasi untuk mengukur performansi dengan membandingkan hasil dari sistem dengan buatan ahli. Berdasarkan [15] nilai F-Measure didapatkan dari perhitungan :

$$- \qquad : \quad \times \frac{\times}{+}$$

Dimana Precision dan Recall adalah

Precision : Bagian dokumen yang relevan yang diambil berdasarkan ringkasan ahli

Recall : Bagian dari dokumen yang relevan dengan threshold yang berhasil diambil

$$: \frac{|\sum \qquad \cap \sum \qquad |}{\sum}$$

$$: \frac{|\sum \qquad \cap \sum \qquad |}{\sum}$$

4. Pengujian

4.1. Nilai Threshold

Agar dapat melihat pengaruh nilai threshold terhadap akurasi dan melihat pergerakan nilai akurasi, nilai threshold dicoba untuk semua kemungkinan dengan kenaikan nilai secara konstan sebesar 0.05. Sehingga akan dilakukan pengujian dengan nilai threshold 0.05, 0.10, 0.15 ….. 0.95.

4.2 Jumlah Data Latih

Pengujian terhadap jumlah data latih dicoba beberapa komposisi, antara lain : 28 (50%) data latih; 33 (60%) data latih; 39 (70%) data latih; 44 (80%) data latih; 50 (90%) data latih.

Kenaikan jumlah data latih dibuat konstan terhadap persentase data agar dapat melihat pengaruh jumlah data latih terhadap nilai akurasi dan melihat pergerakan nilai akurasi tersebut. Untuk data uji digunakan 5 dokumen dipilih secara acak diambil dari setiap subtopik.

4.2. Pengaruh Fitur

Pengujian dilakukan berdasarkan penggunaan fitur

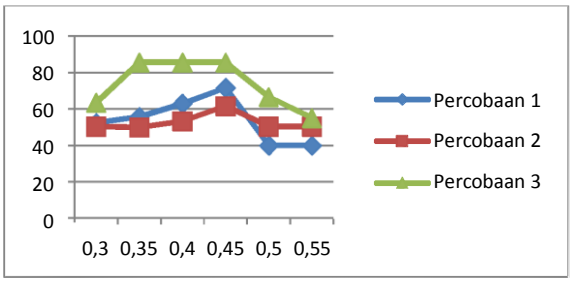
$$( ) = \sum_{\in} ( ) \quad (2.21)$$

yang dimiliki oleh setiap kalimat. Fitur dikelompokkan menjadi 2 jenis, yaitu Basic Feature

dan Complex Feature. Untuk pengujian feature dilakukan pengujian, diantaranya : menggunakan Basic feature saja; menggunakan Basic feature dan LSA_Score; menggunakan Basic feature dan HITS_Score; menggunakan Basic feature dan Complex feature.

4.3. Analisis Hasil Pengujian

Tabel 4-1 Hasil Pengujian Nilai Theshold Terhadap Jumlah Data Latih



Pengujian dilakukan dengan mengkombinasikan semua nilai threshold dengan jumlah data latih dan dilakukan sebanyak tiga kali. Hal tersebut dilakukan untuk menentukan nilai threshold yang paling tepat agar sistem menghasilkan akurasi yang paling baik. Dari Tabel 4-2 dapat dilihat bahwa dari enam kolom percobaan didapatkan nilai threshold lebih sering berada dinilai 0,45 hal tersebut terjadi karena nilai threshold didapatkan dengan melihat nilai precision dan recall yang paling tinggi berdasarkan data keluaran sistem. Berdasarkan pengujian untuk kasus ini untuk mendapatkan akurasi sistem yang paling baik adalah dengan menentukan nilai threshold kalimat summary yaitu sebesar 0.45. Angka tersebut menyatakan batas untuk menentukan kalimat tersebut merupakan kalimat summary jika memiliki nilai P(1|X)>= 0.45 dimana P(1|X) adalah probabilitas kalimat tersebut berlabel 1. Namun berdasarkan percobaan yang dilakukan secara berulang didapatkan nilai akurasi yang berbeda, hal tersebut dikarenakan adanya aspek pelabelan secara manual yang dilakukan oleh tenaga ahli.

4.4. Hasil jumlah data latih

Dengan menerapkan nilai threshold kalimat summary yang didapatkan pada pengujian sebelumnya yaitu 0.45 selanjutnya dilakukan pengujian untuk melihat pengaruh jumlah data latih terhadap akurasi yang dihasilkan oleh sistem. Berdasarkan tabel 4-2 didapatkan hasil pengujian yang bisa dilihat pada Tabel 4-4.

Tabel 4-2 Hasil Pengujian Jumlah Data Latih Dengan Threshold 0.45

Jumlah Data Latih	Akurasi Pengujian 1	Akurasi Pengujian 2	Akurasi Pengujian 3	Akurasi Rata-rata (%)
50% (28 dokumen)	33,33	30,77	30,77	31,63
60% (33 dokumen)	33,33	40	40	37,78
70% (38 dokumen)	40	46,15	42,86	43
80% (44 dokumen)	66,67	57,14	50,00	57,94
90% (50 dokumen)	71,43	61,54	62,50	62,5

Pengujian dilakukan dengan menggunakan beberapa kombinasi data latih yang perubahannya konstan terhadap persentasenya. Dari hasil pengujian yang dilakukan, didapatkan nilai akurasi yang rendah pada penggunaan persentase jumlah data latih yang kecil.

Hal tersebut terjadi karena model yang dibentuk dari jumlah data latih yang kecil menghasilkan model yang tidak akurat karena tidak mencakup variasi dari semua kemungkinan nilai yang ada pada data uji.

4.5. Hasil komposisi feature

Dengan menerapkan nilai threshold kalimat summary yang didapatkan pada pengujian sebelumnya yaitu 0.45 dan menggunakan data latih sebanyak 90% dari jumlah dataset yang ada.

Dilakukan pengujian dengan kombinasi diatas menggunakan 50 data latih dan 5 data uji berdasarkan hasil dari pengujian jumlah data latih di skenario pengujian sebelumnya. Dari hasil pengujian terhadap penggunaan fitur Basic Feature dan Complex Feature menunjukkan nilai akurasi yang besar pada penggunaan complex feature khusunya HITS, sedangkan nilai akurasi rendah pada penggunaan basic feature dan LSA.

Penggunaan complex feature berpengaruh besar pada nilai akurasi karena selain memperhatikan kalimat secara statistikal, complex feature melihat keterkaitan antar kalimat yang ada pada 1 dokumen secara keseluruhan. Pada penggunaan HITS

memiliki pengaruh paling besar pada nilai akurasi karena HITS menilai tingkat kepentingan suatu kalimat dengan berdasarkan graf keterhubungan kalimat yang menggunakan directed backward sehingga memperhatikan urutan kalimat tersebut, dan juga bobot keterhubungan antar kalimat dinilai dari similaritas antar satu kalimat dengan seluruh kalimat pada dokumen.

Penggunaan fitur HITS memiliki pengaruh yang lebih besar pada hasil akurasi dibandingkan dengan fitur LSA, karena pada dasarnya HITS dan LSA memberi nilai pada tingkat kepentingan suatu kalimat pada suatu dokumen. Karena, fitur HITS juga memperhatikan letak dari kalimat tersebut. Contoh, apabila kalimat pertama memiliki nilai similaritas dengan kalimat ke 5 dan kalimat ke 6, maka diartikan kalimat ke 5 dan kalimat ke 6 merujuk ke kalimat pertama. Hal ini didukung dengan analisa bahwa kalimat ringkasan memiliki kemungkinan lebih besar untuk berada di awal dokumen dibandingkan di akhir dan di tengah.

Hasil akurasi yang diperoleh dari penggunaan basic feature dan HITS memiliki akurasi yang sama dengan penggunaan basic feature dan complex feature , hal tersebut dikarenakan fungsi dari nilai LSA sudah dapat diwakilkan/direpresentasikan oleh nilai HITS. Sehingga kalimat yang dianggap penting oleh fitur HITS sama dengan kalimat yang dianggap penting oleh fitur LSA, dan HITS memperbaiki hasil LSA.

Contoh pada gambar dibawah ini, menunjukkan bahwa kalimat yang memiliki nilai tinggi pada fitur HITS juga memiliki nilai yang tinggi pada fitur LSA.

id_kal	id_doc	kalimat	lsa_score	hits_score	label
855	46	Presiden FIFA Sepp Blatter sangat sedih mendengar ...	4.3589	1	1
253	14	Los Blancos juga melepaskan empat ancaman ke gawan...	2.82843	0.682366	0
729	38	Selalu ada persaingan besar bila bicara Piala Duni...	3	0.681773	0
52	4	Gelandang Olympique Marseille Mario Lemina tanpa r...	3.31663	0.575842	1
739	39	Mungkin kata itu pantas disematkan untuk Timnas Pr...	3.74166	0.514742	1
409	22	Setiap anak di dunia memiliki mimpi pun demikian h...	2.82843	0.50075	0
232	13	Badan Sepakbola Eropa (UEFA) akan menyampaikan pes...	3.60555	0.496924	1

Gambar 4.1 Pengaruh HITS Terhadap LSA

Sedangkan HITS menunjukkan bagaimana memperbaiki LSA, dimana saat LSA memberikan nilai tinggi pada kalimat bukan summary, HITS memberikan nilai 0 pada kalimat tersebut ditunjukkan pada gambar dibawah ini

id_kal	id_doc	kalimat	lsa_score	hits_score	label
783	41	Berikut 32 kontestan Piala Dunia 2014 Brasil: Tim ...	22.6063	0	0
764	40	Mereka adalah; Henrikh Mkhitaryan (Armenia) David ...	6.16441	0	0
618	33	Indra bersama seluruh asistennya: Eko Purdjianto N...	6	0.0271629	0

Gambar 4.2 Pengaruh LSA Terhadap HITS

Dari hasil yang didapatkan menunjukkan nilai akurasi yang rendah pada penggunaan basic feature saja, karena basic feature hanya merepresentasikan nilai statistikal dan linguistik dari kalimat tersebut. Dimana nilai nilai tersebut hanya memperhatikan posisi dari suatu kalimat, jumlah kata pada kalimat, dan nilai nilai lain yang merepresentasikan kalimat secara luas.

5. Kesimpulan

1. Metode Conditional Random Fields dapat diimplementasikan untuk membangun sistem peringkasan multi-dokumen dan sistem berhasil menghasilkan ringkasan yang dapat membantu pencarian informasi berita sehingga pencarian informasi menjadi lebih efektif dan efisien.
2. Berdasarkan hasil percobaan yang dilakukan sebanyak 3 kali dapat disimpulkan bawah nilai threshold terbaik didapatkan pada nilai threshold 0,45. Hal tersebut didasarkan bahwa pada nilai threshold tersebut menghasilkan nilai F-Measure yang cukup tinggi. Namun nilai threshold tersebut tidak menjadi acuan nilai terbaik dikarenakan adanya perubahan nilai F-Measure pada setiap perulangan percobaan. Meskipun terdapat perbedaan nilai F-Measure disetiap percobaan namun nilai yang dihasilkan tetap memiliki nilai yang cukup tinggi. Selain itu, nilai threshold 0,45 juga menghasilkan nilai basic feature dan complex feature yang cukup baik. Percobaan yang berulang pada basic feature dan complex feature pun mengalami perubahan nilai namun dengan nilai threshold 0,45 masih menghasilkan nilai yang cukup baik. Adapun penyebab perbedaan nilai F-Measure, basic feature dan complex feature disetiap percobaan disebabkan olek aspek pelabelan yang masih dilakukan secara manual.

6. Saran

1. Menambahkan referensi ringkasan yang dilakukan oleh tenaga ahli lain untuk membandingkan pengaruh ringkasan manual terhadap akurasi.
2. Membangun sistem peringkasan dokumen yang dapat menangani dokumen yang mengandung kata disingkat atau kata tidak baku.
3. Melakukan percobaan dengan menggunakan topik lain untuk melihat pengaruh topik terhadap pembentukan model.

7. Daftar Pustaka

[1] Shen, Dou, Jian-Tao Sun, Hua Li, Qiang Yang, and Zheng Chen. “Document Summarization using Conditional Random Fields.” IJCAI’07 Proceedings of the 20th international joint conference on Artificial intelligence,2007: 2862-2867.

[2] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," 2004.

[3] A. Celikyilmaz and D. Hakkani-Tur, "A Hybrid Hierarchical Model for Multi-Document Summarization," Proceeding of the 48th Annual

Meeting of the Association for Computational Linguistic, pp. 815-824, 2010.

[4] Y. Ouyang, W. Li, S. Li and Q. Li, "Applying regression models to query-focused multi-document summarization," Information Processing and Management, pp. 227-237, 2010.

[5] Lafferty, J., McCallum, A., Pereira, F."Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. "Proc. 18th International Conf. on Machine Learning. Morgan Kaufmann.,2001

[6] Wallach, Hanna M. Conditional Random Fields: An Introduction. Technical Report MS-CIS-04-21, University of Pennsylvania, 2004, 2004.

[7] Lin, Chin-Yew, and Eduard Hovy."Automatic evaluation of summaries using n-gram co-occurrence statistics." NAACL, 2003: 71-78.

[8] B. Susanto, "lecturer.ukdw.ac.id," 5 Februari 2015. [Online]. Available: http://lecturer.ukdw.ac.id/budsus/pdf/textwebmining/TextMining_Kuliah.pdf.

[9] X. Zhu, "Conditional Random Fields," Advanced Natural Language Processing, 2010.

[10] Galanis. Dimitrios, Lampouras. Gerasimos and Androutsopoulos. Ion, "Extractive Multi-Document Summarization with Integer Linear Programming and Support Vector Regression". COLING 2012, pp.911-926, 2012.

[11] A. Thomo, "Latent Semantic Analysis (Tutorial)," 2009.

[12] T. T. Truyen dan D. Phung, "A Practitioner Guide to Conditional Random Fields for Sequential Labelling," Curtin University of Technology, 2008.

[13] M. Benzi, E. Estrada dan C. Klymko, "Ranking Hubs And Authorities Using Matrix Functions" dalam Linear Algebra and its ApplicationsVolume 438,Issue5, Pages 2447–2474, 2013.

[14] R. Mihalcea, "Language Independent Extractive Summarization," dalam Interactive poster and demonstration sessions Pages 49-52 , 2005.

[15] Y. Gong dan X. Liu, "Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis," dalam SIGIR '01 Pages 19-25 , 2001.